This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICCE56470.2023.10043585 2023 IEEE International Conference on Consumer Electronics (ICCE)

Circuit Design of Memristor-based GRU and its Applications in SOC Estimation

Jiayang Wang
School of Electronics & Information
Hangzhou Dianzi University
Hangzhou, China
w.jy@hdu.edu.cn

Xiaoyue Ji
College of Electral Engineering
Zhejiang University
Hangzhou, China
ji.xiaoyue@zju.edu.cn

Zhekang Dong
School of Electronics & Information
Hangzhou Dianzi University
Hangzhou, China
englishp@hdu.edu.cn

Zhaowei Zhang
School of Electronics & Information
Hangzhou Dianzi University
Hangzhou, China
zzw0211@hdu.edu.cn

Chun Sing Lai*
Departmrnt of Electronic & Computer Engineering
Brunel University London
London, UK
chunsing.lai@brunel.ac.uk

Donglian Qi
College of Electral Engineering
Hainan Institute of Zhejiang University
Sanya, China
qidl@zju.edu.cn

***Abstract*—Gated recurrent unit (GRU) is a variant of recurrent neural network (RNN), which is widely used in applications and tasks related to sequence data processing. However, traditional GRU networks based on von Neumann computing construction, have been facing challenges such as big data dimension and high real-time requirements. Meanwhile, limited internal storage resources and external storage bandwidth have jointly limited the overall performance of hardware implementation of GRU networks. Based on these, a compact scheme for the hardware circuit design of memristor-based GRU network is presented, along with the concrete circuit design of nonlinear activation and the linear matrix operation. The entire scheme is validated by the application of the Lithium ion battery state of charge (SOC) estimation. This work is expected to integrate the neuromorphic electronics and battery management systems, and further promote the development of electric vehicles in smart cities.**

Keywords—gated recurrent unit (GRU), hardware circuit design, memristor crossbar array, state of charge (SOC)

I. Introduction

Recurrent Neural Network (RNN) is a kind of neural network with short-term memory ability, which has been widely used in control systems, action recognition, recommendation systems, speech identification, sentiment classification and other sequence data processing problems [1-5]. However, RNN can only learn the dependencies between short-time series in practice due to the gradient vanishing or explosion caused by repeated multiplication of cyclic weight matrices in the training process [6]. Therefore, some variants of RNN, such as the long short-term memory (LSTM) [7] and the gated recurrent unit (GRU) [8], have been proposed to handle the problem of insensitivity to long-term dependencies.

Both LSTM and GRU enforce a constant error flow over time steps and use gates on the input and the recurrent input to regulate the information flow through the network [9]. Different with LSTM, GRU has a more simple structure with less parameters, but reaches approximate accuracy as the LSTM or even outperforms it in some application cases [10]. Considering the design architecture, energy consumption and time delay, this work mainly focuses on the hardware implementation of GRU.

The time series analysis of traditional GRU is mostly performed on CPUs and GPUs, which still suffers from some limitations, such as the large data dimensions and high real-time requirements [11]. Meanwhile, the gates utilized in the GRU determines that the computations are not parallel, thus increasing the computational complexity and resource consumption especially in some complicated tasks. The hardware implementation of GRU can be deemed as an effective remedy for addressing the research gaps in computation complexity and energy consumption. So far, a few attempts have been made to implement the GRU hardware circuit, except for the field programmable gate array (FPGA)-based implementations [12, 13]. Reference [12] presented an accelerator for matrix vector multiplications in GRU network mapped to Stratix V and Arria 10 FPGAs as well as a 14-nm ASIC. Compared to the software implementations on CPUs and GPUs, this hardware architecture offers significant efficiency improvements. Reference [13] proposed a hardware implementation of GRU using VHDL on the Altera Arria 10 GX FPGA, which shows a significantly high performance per unit power.

In account of the limited internal storage resources and external storage bandwidth, the FPGA-based GRU implementations are difficult to achieve efficient training [14]. Particularly, the cost of computing deployment on FPGA platform is relatively high. How to find an effective circuit component and use it for the hardware implementation of GRU is urgent and necessary.

As the forth basic circuit element, memristor has superior non-volatile properties, ultra-low power consumption, nanoscale structure, and ease of integration [15]. Furthermore, organizing a large number of memristors in a crossbar array can achieve in-memory parallel computing with very low power consumption. So far, there have been a wide variety of neuromorphic computing systems have been proposed using memristive circuits [16-18]. But to our knowledge, this work first proposes the GRU hardware construction using memristive circuits and further applies it to one of the potential consumer electronics applications, i.e., the state-of-charge (SOC) estimation of Lithium ion battery.

This paper proposes a novel hardware circuit framework of the GRU via memristive circuits. With such design concept and deep learning integration, the proposed memristor-based GRU can be used to realize real-time Lithium ion battery SOC estimation, guaranteeing the durability and reliability of the entire electric vehicle system. The main contributions of this paper are summarized below:

This work is supported by National Natural Science Foundation (NNSF) of China under Grant No. 62001149, Zhejiang Provincial Nature Science Foundation of China under Grant No. LQ21F010009, and Fundamental Research Funds for the Provincial Universities of Zhejiang under Grant No. GK229909299001-06.

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICCE56470.2023.10043585 2023 IEEE International Conference on Consumer Electronics (ICCE)

(1) Different from the existing FPGA-based GRU implementations, a circuit design of memristor-based GRU with stronger real-time performance is proposed.

(2) The proposed memristor-based GRU network is used to realize the SOC estimation of Lithium ion battery. The experimental results demonstrate its superior performance compared to several existing competitors in the estimation accuracy and temperature sensitivity. Notably, it is the first attempt to use the hardware-based GRU in the field of sequence estimation.

The remainder of this paper is organized as follows. Section II briefly describes the memristor model and details the corresponding crossbar array. In Section III, the specific circuit design of memristor-based GRU is proposed. For the verification purpose, the presented network is applied to implement the SOC estimation of Lithium ion battery in Section IV. Finally, Section **Error! Reference source not found.** concludes the entire work.

II. Memristor Crossbar Array

A. Memristor Model and Performance Analysis

Since Hewlett-Packard (HP) Laboratory proved the existence of memristor in 2008, a variety of memristor models have been devised over the last years, such as the nonlinear ion drift model, TSSM memristor model, Simmons Tunnel Barrier model, and so forth [15, 19, 20]. Among them, the Voltage ThrEshold Adaptive memristor (VTEAM) model is simple, versatile, accurate and allows simulating the basic characteristics of numerous voltage-controlled memristor models by adjusting the model parameters [21]. Here, the VTEAM model is chosen for analog simulations in this paper.

Specifically, the principal equations of the VTEAM model can be written by:

$$\frac{dx}{dt}=\begin{cases} k_{\text{off}}\left(\dfrac{v(t)}{v_{\text{off}}}-1\right)^{\alpha_{\text{off}}} f_{\text{off}}(x), 0<v_{\text{off}}\le v \\ 0, v_{\text{on}}<v<v_{\text{off}} \\ k_{\text{on}}\left(\dfrac{v(t)}{v_{\text{on}}}-1\right)^{\alpha_{\text{on}}} f_{\text{on}}(x), v\le v_{\text{on}}<0 \end{cases} \tag{1}$$

$$i(t)=\left[R_{\text{on}}+\frac{R_{\text{off}}-R_{\text{on}}}{x_{\text{off}}-x_{\text{on}}}(x-x_{\text{on}})\right]^{-1}\cdot v(t) \tag{2}$$

$$f_{\text{off}}(x)=\exp\left[-\exp\left(\frac{x-a_{\text{off}}}{w_{\text{c}}}\right)\right] \tag{3}$$

$$f_{\text{on}}(x)=\exp\left[-\exp\left(-\frac{x-a_{\text{on}}}{w_{\text{c}}}\right)\right] \tag{4}$$

where k_{on}, k_{off}, α_{on}, α_{off}, a_{on}, a_{off}, w_{c} are the fitting parameters. R_{on}, R_{off} and x_{on}, x_{off} are the limit values of memristance and the state variable x in the ON and OFF regime. v_{on}, v_{off} are the threshold voltages. $v(t)$ denotes the voltage applied to the memristor model, and $i(t)$ is the corresponding current passing through the device. $f_{\text{on}}(x)$ and $f_{\text{off}}(x)$ are the so-called window functions, they are used to guarantee the state variable x is always within the range of $[x_{\text{on}}, x_{\text{off}}]$.

Furthermore, the performance analysis of the VTEAM model is conducted by a series of computer simulations (MATLAB 2018b platform). The necessary parameter setting is provided in Table I:

TABLE I. Memristor Parameters of Vteam Model

Parameters	Value	Parameters	Value
k_{on}	-1	k_{off}	1
α_{on}	0.1	α_{off}	0.1
a_{on}	1.8e-9	a_{off}	1.2e-9
R_{on}	100Ω	R_{off}	16000Ω
x_{on}	0	x_{off}	3e-9
w_{c}	107e-12	x_0 [a]	0

[a] x_0 is the initial state variable of memristor.

The corresponding simulation results are provided in Fig. 1:

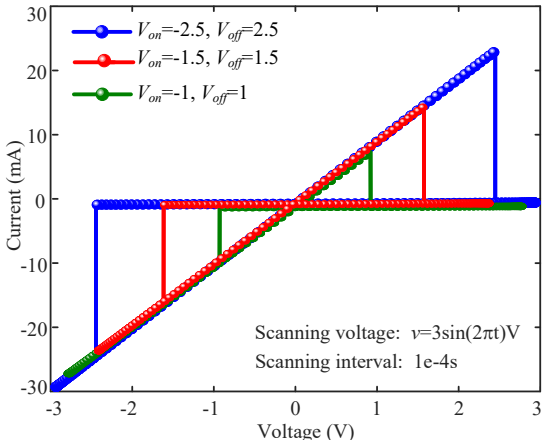

Fig. 1. The V-I curves of the VTEAM model

Fig. 1 exhibits the voltage-current (V-I) curves of the VTEAM model with diverse threshold voltages (i.e., $V_{\text{on1, off1}}$=±1V, $V_{\text{on2, off2}}$=±2V, and $V_{\text{on3, off3}}$=±2.5V). It is clear that all the V-I curves show the hysteresis loops with different amplitude levels. When the scanning voltage satisfies $v \ge v_{\text{off}}$, the memristance will increase to its highest value R_{off}. Similarly, when the voltage across the device satisfies $v \le v_{\text{on}}$, the memristance will rapidly decrease to its lowest value R_{on}. In other cases, the memristance remains unchanged .

B. Memristor Crossbar Array

Memristor crossbar array offers nonvolatile resistance states and could break the energy and speed limitation in vector-matrix multiplication operation [22]. Nevertheless, considering memristor crossbar array still suffers from the inevitable sneak-path current issue, which may cause cross-talk interference between adjacent memory cells and result in misinterpretation during the writing/reading process. In this paper, one transistor and one memristor (1T1M) structure is built to isolate unselected rows to avoid this issue.

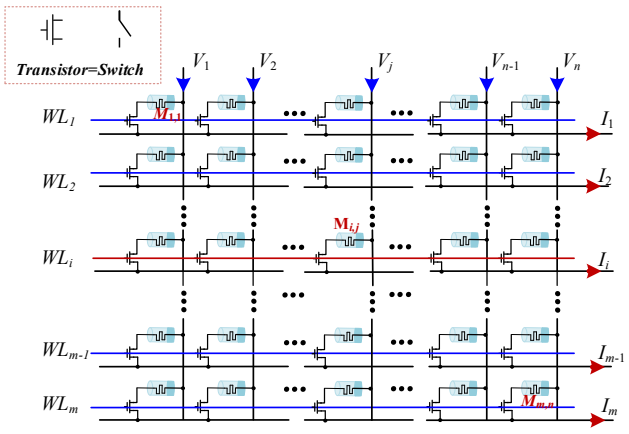

Fig. 2. The 1T1M crossbar array

In 1T1M crossbar array, the control signal (in the form of voltage) applied to the word line controls the state of the memristor. When the control signal is in the low level, no current flows through the memristor. While when the control signal is in the high level, there exists a current pathing through the memristor, the output current of the architecture I_{out} can be described as $I_{out}=\sum W_{ij} \cdot V_{in}$, where $W_{i,j}$ is the conductance of the memristor located in (i, j). V_{in} is the input voltage applied to the bit line.

III. Circuit Design of Memristor-Based GRU

A. GRU Network

GRU aims to solve the problems of gradient explosion and disappearance in RNNs while maintaining the long-term information [8]. A typical GRU cell concludes two components, i.e., the update gate and the reset gate. The specific internal structure is illustrated in Fig. 3.

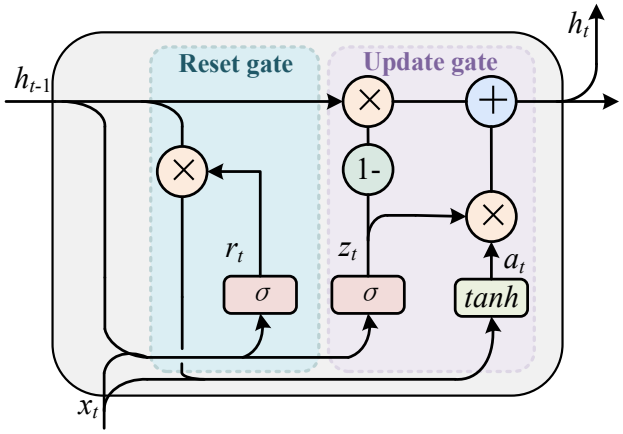

Fig. 3. The internal strucure of GRU

In Fig. 3, GRU consists of the present input x_t, the previous hidden state h_{t-1}, the reset gate r_t, the update gate z_t, the candidate output state a_t, and the new hidden state h_t. The reset gate filters out previously irrelevant information in the hidden layer, and the update gate determines how much past information can continue to be passed to the future [8]. The internal operating mechanism of GRU can be expressed by:

$$\begin{bmatrix} r_t \\ z_t \end{bmatrix} = \begin{bmatrix} W_r & U_r & b_r \\ W_z & U_z & b_z \end{bmatrix} \begin{bmatrix} x_t \\ h_{t-1} \\ 1 \end{bmatrix} \tag{5}$$

$$a_t = \tanh\left(W_h x_t + U_h\left[h_{t-1} \odot \sigma\left(r_t\right)\right] + b_h\right) \tag{6}$$

$$h_t = \left[1-\sigma\left(z_t\right)\right] \odot h_{t-1} + \sigma\left(z_t\right) \odot a_t \tag{7}$$

where W_r, W_z, W_h, U_r, U_z, U_h, b_r, b_z, and b_h are all the network parameters. σ denotes the logistic sigmoid function, it is used to constrain the data within the bounds of [0, 1]. $\odot$ is Hadamard Product, representing the element-wise multiplication.

B. Circuit Design

Based on (5), (6) and (7), the entire circuit design of GRU network is provided in Fig. 4. The specific description is provided below.

From Fig. 4, the entire circuit system contains many sub-circuits, for example, the 1T1M crossbar array (labeled by pale yellow box and mainly for matrix-vector multiplication), the add circuit (labeled by green box), the V-I transfer circuit (labeled by purple box), the activation function circuit (labeled by grey box), the subtraction circuit (labeled by the blue box), and the multiplication circuit (labeled by red box and mainly for scalar multiplication). All these circuits are integrated to effectively simulate (5), (6), and (7). Here, we mainly discuss the activation circuit, and the other sub-circuits are all mature circuits that have been used in many other fields. Thus, we will not repeatedly describe them in this part.

Unlike the other activation circuits [23], the activation circuit in GRU network needs to realize both the logistic sigmoid function and the hyperbolic tangent function. Meanwhile, considering the integrity of the entire system, both the input and output of the activation circuit should be in the voltage form. Based on these, a suitable design scheme with a single memristor is presented in Fig. 5. The

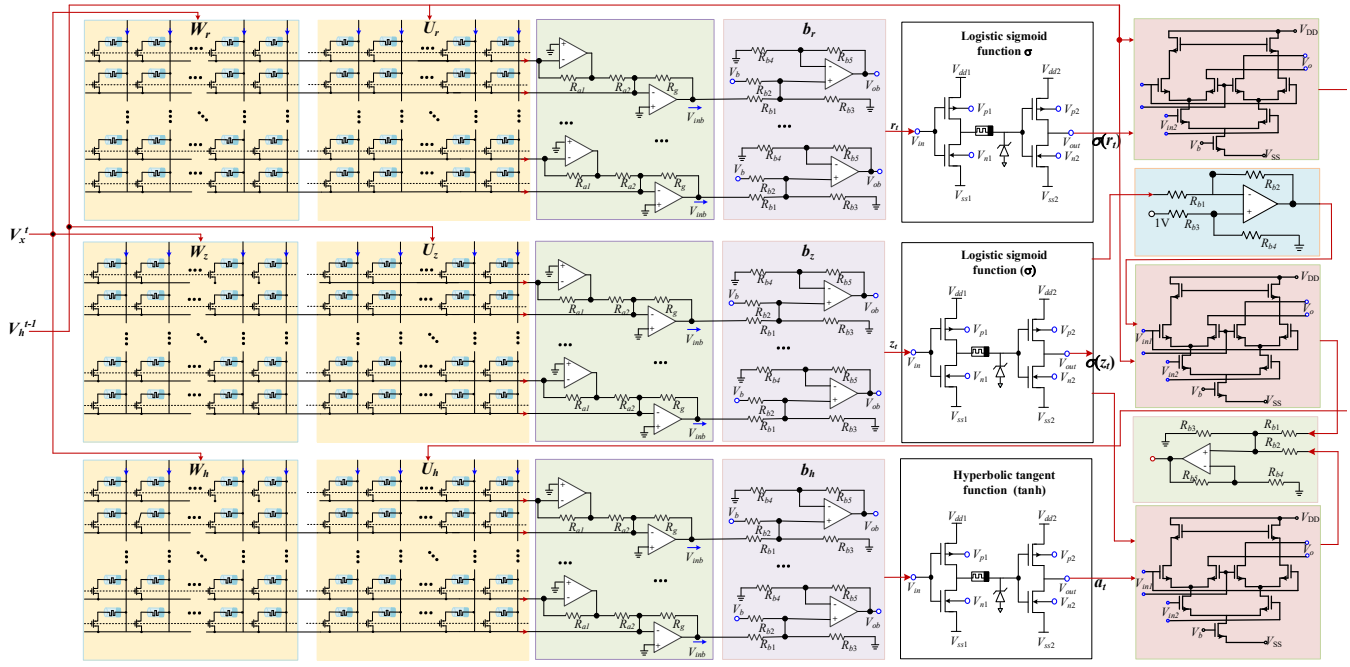

Fig. 4. The circuit design of GRU network using memristor circuits

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICCE56470.2023.10043585 2023 IEEE International Conference on Consumer Electronics (ICCE)

corresponding circuit parameter setting is provided in Table II to obtain hyperbolic tangent function.

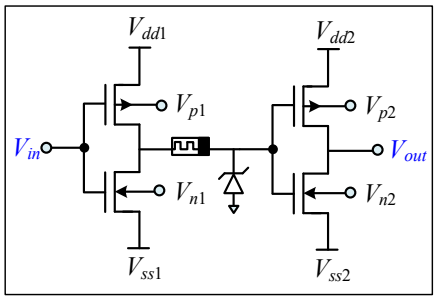

Fig. 5 The activation function circuit with a single memristor

TABLE II. PARAMETERS OF ACTIVATION FUNCTION CIRCUIT

Parameters	Value (V)	Parameters	Value (V)
V_{dd1}	1.3	V_{dd2}	1.3
V_{p1}	0.4	V_{p2}	0.5
V_{n1}	-1.2	V_{n2}	-0.4
V_{ss1}	-.05	V_{ss2}	-1.1

IV. MEMRISTOR-BASED GRU FOR SOC ESTIMATION

Nowadays, the efficient, long-life, economic and environmental-friendly Lithium ion battery has become the major resources in the electric vehicles [25]. An accurate estimation of the SOC of a battery system is actually an effective remedy to break the 'range anxiety' bottleneck. Based on this, the proposed system is applied to the SOC estimation of Lithium ion battery for the purpose of verification.

The error backpropagation is conducted on a workstation (CPU Intel Core i7-10700K, 32.0GB RAM), and the proposed GRU network is used for forward computation. The necessary parameter setting: the circuit parameters (containing V_{on}, V_{off}, V_{WL}, V_{BL}, V_{read}, and scan rate) are set to -2V, 2V, 3V, 1.7V, and 0.05V/s respectively; The network parameters (containing learning rate, momentum, and epochs) are set to 0.01, 0, and 128 respectively.

Meanwhile, this work uses the public Lithium ion battery dataset collected from the Samsung 18650 $LiNiMnCoO_2$/Graphite Lithium ion batteries by University of Maryland [26], including the current and voltage data of BJDST, DST, FUDS and US06 drive cycles in varying ambient temperatures. Following [25], the random mixture of BJDST, DST and US06 is used as a training set, FUDS is used as a test set.

Fig. 6 shows the SOC estimation results using the proposed method at 0°C, 25°C and 45°C with an initial SOC of 80%. The validity of the proposed system is demonstrated by comparing the predicted SOC values (blue line) with the real SOC values (black line). In addition, it can be easily found from the human visual system that the proposed system has relatively poor SOC estimation accuracy in low temperature environment.

Furthermore, to evaluate the overall performance of the proposed system, a comparative study is performed between the proposed model and some existing SOC estimation methods, including Bi-GRU, LSTM-PF, SVM-PF, BPNN-BSA, ELM-BSA, EKF, UKF [25]. Here, the Root Mean Square Error (RMSE) is applied to carry out the objective analysis. Its mathematical expression is given by:

$$RMSE = \frac{1}{T}\sqrt{\sum_{t=1}^{T}(SOC_t - SOC_t^{'})} \quad (8)$$

where SOC_t and $SOC_t^{'}$ are the actual and estimated SOC at time t, respectively. T is the total number of the test data. Commonly, smaller RMSE indicates better fitting effect.

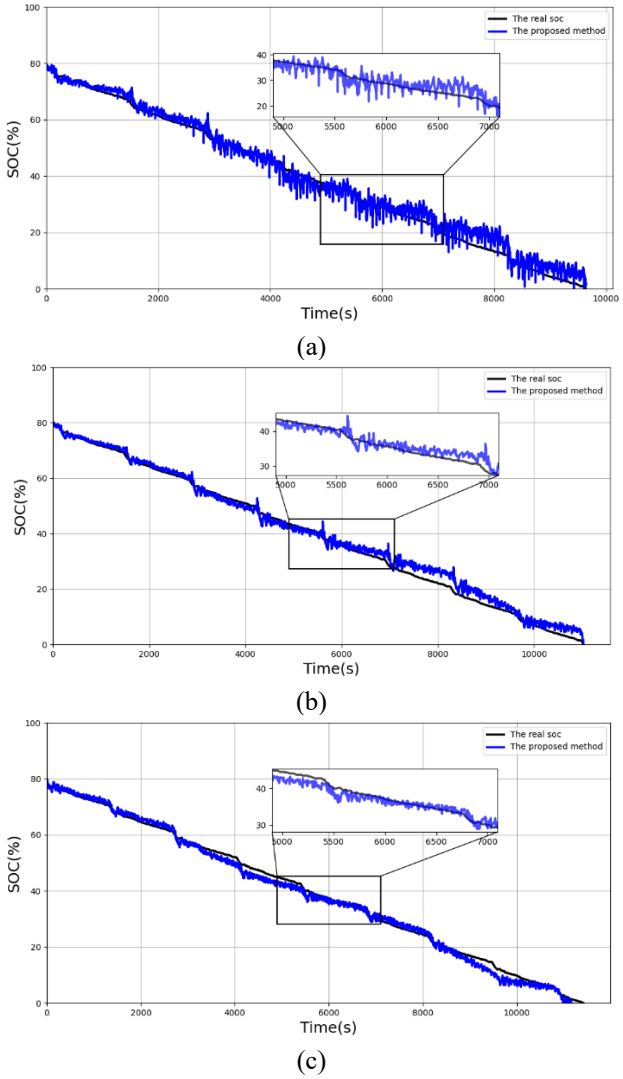

Fig. 6 Estimation results on the FUDS test using different estimation methods at ambient temperatures: (a) 0°C, (b)0°C, and (c) 45°C

The comparative information (RMSE) of various methods on FUDS under various ambient temperatures is collected in Table III.

TABLE III. COMPARATIVE INFORMATION OF VARIOUS METHODS

SOC Estimation Methods	Ambient Temperature		
	0°C	25°C	45°C
Bi-GRU	2.15	1.30	1.34
LSTM-PF	1.42	1.50	1.68
SVM-PF	2.97	1.65	2.24
BPNN-BSA	1.74	0.91	0.57
ELM-BSA	3.72	2.34	2.27
EKF	/	4.31	5.70
UKF	/	3.97	3.08
The proposed method	**2.18**	**1.36**	**1.23**

From Table III, it is clear that ambient temperatures have a great influence on SOC estimation accuracy, especially in low ambient temperatures. It is observed that for FDUS cycle, the LSTM-PF model, which has good performance in terms of accuracy and stability in varying temperatures, achieving the value of RMSE as 1.42%,1.50% and 1.68% respectively at 0℃, 25℃ and 45℃. For BPNN-BSA method, the RMSE is also low, which is 1.74%, 0.91% and 0.57% respectively. However, the SOC estimation results are easier affected by the varying temperatures. The EKF and UKF are filtering-based methods, these kind of methods always have higher RMSEs in account of the simple model structure and the deficient datasets. Compared with these competitors, the proposed system has a relatively high estimation accuracy and good stability, proving the effectiveness of the proposed system in SOC estimation of Lithium ion battery. Especially, due to the advantages of the running speed, the proposed method can realize the entire SOC estimation within a very short time, indicating its advantages in efficiency and cost (mainly refers to the time cost).

V. Conclusions

This paper mainly focus on the investigation of circuit design of hardware memristor-based GRU network. Specifically, the electrical characteristics of VTEAM is examined through detailed formula derivation and numerical simulations. Then, the memristor circuit with the crossbar array configuration is realized, achieving in-memory parallel computing with high efficiency. Correspondingly, the compact memristor-based GRU network is proposed, along with the concrete circuit design of activation function circuit and other functional circuits. For the purpose of verification, the presented memristor-based GRU network is applied to the SOC estimation of Lithium ion battery. The experimental results demonstrate that the proposed scheme has relatively good estimation accuracy and stability under the three temperatures of 0℃, 25℃ and 45℃.

References

[1] A. Sarabu and A. K. Santra, “Human action recognition in videos using convolution long short-term memory network with spatio-temporal networks,” Emerg. Sci. J., vol. 5, no. 5, pp. 25–33, February 2021.

[2] F. Karim, S. Majumdar, and H. Darabi, “Multivariate LSTM-FCNs for time series classification,” Neural Netw., vol. 116, pp. 237-245, August 2019.

[3] K. Karim et al, “Human action recognition using attention based LSTM network with dilated CNN features,” Future Gener. Comput. Syst., vol. 125, pp. 820-830, December 2021.

[4] J. Siłka, M. Wieczorek, and M. Woźniak, “Recurrent neural network model for high-speed train vibration prediction from time series,” Neural Comput. Appl., vol. 34, no. 3, pp. 1-14, August 2022.

[5] T. Donkers and B. Loepp, and J. Ziegler, “Sequential user-based recurrent neural network recommendations,” Proceedings of the eleventh ACM conference on recommender systems, pp. 152-160, August 2017.

[6] J. M. Ackerson, R. Dave, and N. Seliya, “Applications of recurrent neural network for biometric authentication & anomaly detection,” Inf., vol. 12, no. 7, pp. 272, September 2021.

[7] S. Hochreiter and J. Schmidhuber, (1997). “Long short-term memory,” Neural Comput., vol. 9, no. 8, pp. 1735-1780, December 1997.

[8] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” arXiv preprint arXiv:1412.3555, vol. 9, no. 8, pp. 1735-1780, December 2014.

[9] S. Li, W. Li, C. Cook, C. Zhu, and Y. Gao, “Independently recurrent neural network (indrnn): Building a longer and deeper RNN,” Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5457-5466, March 2018.

[10] A. Sherstinsky, “Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network,” Physica D. vol. 404, pp 1-16, March 2020.

[11] Z. Cao et al, “Improving prediction accuracy in LSTM network model for aircraft testing flight data,” 2018 IEEE International Conference on Smart Cloud (SmartCloud), pp. 7-12, September 2018.

[12] Z. S. Zaghloul and N. Elsayed, “The FPGA Hardware Implementation of the Gated Recurrent Unit Architecture,” SoutheastCon, pp. 1-5, March 2021.

[13] E. Nurvitadhi, J. Sim, D. Sheffield, A. Mishra, S. Krishnan, and D. Marr, “Accelerating recurrent neural networks in analytics servers: Comparison of FPGA, CPU, GPU, and ASIC,” 26th International Conference on Field Programmable Logic and Applications (FPL), pp. 1-4, August 2016.

[14] Z. Que et al, “Optimizing reconfigurable recurrent neural networks,” IEEE 28th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), pp. 10-18, May 2020.

[15] D. B. Strukov, G. S. Snider, D. R. Stewart and R. S. Williams, “The missing memristor found,” Nature, vol. 453, no. 7191, pp. 80-83, June 2008.

[16] Y. Ran et al, “A review of Mott insulator in memristors: The materials, characteristics, applications for future computing systems and neuromorphic computing,” Nano Res., pp.1-18, August 2022.

[17] S. Kumar, X. Wang, J. P. Strachan, Y. Yang, and W. D. Lu, “Dynamical memristors for higher-complexity neuromorphic computing,” Nat. Rev. Mater., vol. 7, no. 1, pp. 1-17, April 2022.

[18] M. Ismail, U. Chand, C. Mahata, J. Nebhen, and S. Kim, “Demonstration of synaptic and resistive switching characteristics in W/TiO2/HfO2/TaN memristor crossbar array for bioinspired neuromorphic computing,” J. Mater. Sci. Technol., vol. 96, pp. 94-102, Januray 2022.

[19] X. Ji, D. Qi, Z. Dong, C. S. Lai, G. Zhou, and X. HU, “TSSM: Three-State Switchable Memristor Model Based on Ag/TiOx Nanobelt/Ti Configuration,” Int. J. Bifurcat. Chaos, vol. 31, no. 07, pp 1-13, April, 2021.

[20] M. Khalid, “Review on various memristor models, characteristics, potential applications, and future works,” Trans. Electr. Electro., vol. 20, no. 4, pp. 289-298, June 2019.

[21] D. Biolek, Z. Kolka, V. Biolková, Z. Biolek, and S. Kvatinsky, “(V)TEAM for SPICE simulation of memristive devices with improved numerical performance,” IEEE Access, vol. 9, pp. 30242-30255, February 2021.

[22] Z. Dong, X. Ji, G. Zhou, M. Gao and D. Qi, “Multimodal Neuromorphic Sensory-Processing System with Memristor Circuits for Smart Home Applications,” IEEE Trans. Ind. Appl., 2022 (Early Access).

[23] T. Yang, T. Wei, Z. Tu, H. Zeng, M. Kinsy, N. Zheng, and P. Ren, “Design space exploration of neural network activation function circuits, ” IEEE Trans. Comput. Aid. D., vol. 38, no. 10, pp. 1974-1978, September 2018.

[24] F. Zheng, Y. Xing, J. Jiang, B. Sun, J. Kim and M. Pecht, “A functional hybrid memristor crossbar-array/CMOS system for data storage and neuromorphic applications,” Appl. Energ., vol. 183, pp. 513-525, December 2016.

[25] Z. Zhang et al, “An improved bidirectional gated recurrent unit method for accurate state-of-charge estimation,” IEEE Access, vol. 9, pp. 11252-11263, Januray 2021.

[26] Y. Chen, , W. Luo, M. Carter, L. Zhou, J. Dai, K. Fu, and L. Hu, “Organic electrode for non-aqueous potassium-ion batteries,” Nano Energ., vol. 18, pp. 205-211, November 2015.